

A2 A new class of microcontroller provides an analog and/or digital subsystem comprising many dynamically configurable blocks. An example of such a device is the CY8C25xxx/26xxx family, commercially available from Cypress MicroSystems, Inc., of Bothell, Washington. These blocks do not, in general, provide the equivalent function of a traditional microprocessor peripheral. They may be thought of as fundamental building blocks. However, these blocks may be combined to create such functions. In further contrast to the fixed peripherals of a traditional microcontroller, these blocks may be recombined in a different combination to perform a different function.

Please replace the paragraph beginning at page 15, line 10 with the following:

A3 System 600 optionally includes a radio frequency module 660, which may implement a variety of wireless protocols, for example IEEE 802.11 or BLUETOOTH™ telecommunication and computer communications (BLUETOOTH is a trademark of Bluetooth SIG, Inc., Washington D.C.).

Please replace the paragraph beginning at page 15, line 18 with the following:

A4 Microcontroller system 300 may contain microcontroller 350. Microcontroller 350 may execute instructions stored within memory elements of microprocessor system 300 to configure configurable elements of microcontroller system 300, including configurable analog blocks 310, configurable digital blocks 320, programmable interconnect 330 and configurable I/O transceivers 340. Configuration information for these configurable elements may have been generated by embodiments of the present invention.

Please replace the paragraph beginning at page 15, line 25 with the following:

A5 cmt Microcontroller system 300 may include configurable analog ("PSOC") blocks 310. "PSOC" is a trademark of Cypress MicroSystems of Bothell, Washington, for a programmable, dynamically configurable microcontroller. Configurable analog blocks 310 may be an array of low level building

A5
cancel. level building blocks designed to implement analog functions. Similarly, configurable digital blocks 320 may be an array of low level building blocks designed to implement digital functions.

~~Please replace the paragraph beginning at page 17, line 22 with the following:~~

A6 Figure 5 shows an exemplary User Module placement display 500, according to an embodiment of the present invention. User Module placement involves the placement of selected User Modules onto the configurable blocks 310 and/or 320 available on the selected device.

~~Please replace the paragraph beginning at page 19, line 16 with the following:~~

A7 The CLOCK parameter 6040 may always map to an area in the upper left corner of the block 6070. Embodiments of the present invention may be configured so that there is only one parameter whose TYPE attribute is set to CLOCK in any one configurable block within any User Module. The CLOCK parameter does not have a text block similar to parameter text area 6060, but shows the triangle symbol within the block 6070.

~~Please replace the paragraph beginning at page 20, line 4 with the following:~~

A8 The INPUT-MUX parameter 6050 is a special case parameter that may cause the line to be drawn between the top row of configurable analog blocks, the ANALOG-CT blocks, and the Pin Input MUX control above them. The INPUT-MUX 6050 parameter does not have a bitfield associated with it, so the BITFIELD attribute should be set to NONE. Consequently, an INPUT-MUX 6050 parameter will not appear in the Parameter Pane. The value shown next to the line is controlled by the Pin Input MUX control above. The INPUT-MUX 6050 parameter may only be used with ANALOG-CT blocks.

~~Please replace the paragraph beginning at page 22, line 16 with the following:~~

According to an embodiment of the present invention, the values of some of the attributes in user module description 700 are controlled by the names used in the Device Description XML files, while others are controlled by the elements and attributes in the User Module XML file itself.

A9

Each User Module XML file may contain the specifications for one, and only one User Module. From the root node <DEVICE_DB> 702, the User Module XML file may contain one-and-only-one <USER_MODULE_LIST> 704 element. The <USER_MODULE_LIST> 704 element can contain a multiplicity of <USER_MODULE> elements, for example <USER_MODULE_LIST> 706, but generally will contain only one. The <USER_MODULE> element may have the following attributes as described below.

Please replace the paragraph beginning at page 23, line 2 with the following:

A10

The NAME attribute is the reference for the User Module used by an editor (configuration editing software). It is also the name that appears in under the icon in the User Module Catalog on the User Module Selection view.

Please replace the paragraph beginning at page 24, line 4 with the following:

A11

The <USER_MODULE> element may contain one, and only one of each of the following elements: <SHAPE>, <PARAMETER_LIST> and <API_REGISTER_ALIAS_LIST>.

Please replace the paragraph beginning at page 24, line 8 with the following:

A12

The <SHAPE> element, for example <SHAPE> element 707, specifies the configurable blocks and the resources required by the User Module, and establishes some reference names for use in other parts of the User Module description. The <SHAPE> element has only a SHAPE_TYPE attribute. The SHAPE_TYPE attribute may be set to BLOCK_LIST.

~~Please replace the paragraph beginning at page 24, line 18 with the following:~~

A13
The `<BLOCK_LIST>` element describes a collection of configurable blocks that are connected between blocks within the `<BLOCK_LIST>` element. When multiple `<BLOCK_LIST>` elements are included within a `<SHAPE>` element, each `<BLOCK_LIST>` is placed on the device's configurable blocks independently. Connections between configurable blocks from distinct `<BLOCK_LIST>` elements can exist through resources identified in the `<RESOURCE_LIST>` element.

~~Please replace the paragraph beginning at page 25, line 10 with the following:~~

A14
The `NAME` attribute is a local name that identifies the block within the User Module description. The `NAME` must be unique within a User Module, but similar names can be used in different User Modules. The `NAME` attribute value appears in the GUI in the User Module Placement view. When a User Module is placed, the configurable blocks onto which the User Module is placed show the instance name of the User Module with the local name directly under it.

~~Please replace the paragraph beginning at page 25, line 17 with the following:~~

A15
The `TYPE` attribute specifies the type of configurable block that the User Module requires. The valid values for the `TYPE` attributes include: `DIGITAL`, `DIGITAL_COMM`, `ANALOG_CT`, `ANALOG_SCA`, `ANALOG_SCB` and `ANALOG_SC`.

~~Please replace the paragraph beginning at page 25, line 22 with the following:~~

A16
Cm't
The difference between this list and the configurable block types used in the device description is the addition of the `ANALOG_SC` type in the User Module description. The `ANALOG_SC` block type indicates that the specified block can be placed on an `ANALOG_SCA` or

A16
contd. an ANALOG_SCB block. Similarly, a DIGITAL block type can be placed on a DIGITAL or a DIGITAL_COMM block.

~~Please replace the paragraph beginning at page 26, line 1 with the following:~~

A17 The INTERRUPT_SOURCE and INTERRUPT_TYPE attributes relate to an interrupt handler associated with the configurable block. If an interrupt handler is not associated with the configurable block, then the INTERRUPT_SOURCE and INTERRUPT_TYPE attributes should not be included in the <BLOCK> element. The INTERRUPT_SOURCE attribute value is a name that is used to generate the interrupt handler name. The INTERRUPT_SOURCE attribute value is appended to the User Module instance name to form the interrupt handler name. The INTERRUPT_TYPE attribute specifies whether a LJMP (long jump) or a CALL (to subroutine) is used when calling the interrupt handler from the vector table.

~~Please replace the paragraph beginning at page 26, line 14 with the following:~~

A18 The <REGISTER_LIST> element, for example <REGISTER_LIST> element 712, may specify the configurable block personalization. The bitfield values set in this element are set on the configurable block where the User Module is located.

~~Please replace the paragraph beginning at page 26, line 18 with the following:~~

A19 The <REGISTER_LIST> consists of one-or-more <REGISTER> elements, for example <REGISTER> element 714. The <REGISTER> element may have a NAME attribute and one <BITFIELD_LIST> element. The NAME attribute value must match one of the names of the registers in a configurable block of similar type contained in the <COMMON_BLOCK_LIST> element of the device description.

~~Please replace the paragraph beginning at page 27, line 5 with the following:~~

The `<INPUT_LIST>` element, for example `<INPUT_LIST> 720` of Figure 7B, specifies the interconnection between blocks and resources within the User Module description. The `<INPUT_LIST>` element has no attributes and consists of one-or-more `<INPUT>` elements. The ~~A20~~ `<INPUT>` element specifies a connection between the block and another block or resource within the User Module description. The relevance of an `<INPUT>` element is that a connection with another configurable block or resource on the device implies that a bitfield within the block registers must be set to a particular value.

~~Please replace the paragraph beginning at page 27, line 14 with the following:~~

The `<INPUT>` element may have the attributes described below. The `SOURCE` attribute may be the name of the resource or configurable block that is the source of the input. The `TYPE` attribute may be an enumeration of the type of the `SOURCE`, either `BLOCK` or `RESERVED_RESOURCE`. The `BLOCK_TYPE` attribute may be an enumeration of the type of the block on which the block is placed, either `ANALOG_SCA` or `ANALOG_SCB`. The `REGISTER_NAME` attribute may be the name of the register within the block `<REGISTER_LIST>` containing the relevant bitfield. The `BITFIELD` attribute may be the name of the bitfield associated with the input.

~~Please replace the paragraph beginning at page 29, line 25 with the following:~~

~~A22~~ The `SOURCE` attribute may be the name of the resource or configurable block that is the source of the input. The `TYPE` attribute may be an enumeration of the type of the `SOURCE`, either `BLOCK` or `RESERVED_RESOURCE`. The `REGISTER_NAME` attribute may be the name of the register within the block `<REGISTER_LIST>` containing the relevant bitfield. The `BITFIELD` attribute may be the name of the bitfield associated with the input.

~~Please replace~~ the paragraph beginning at page 30, line 24 with the following:

The <PARAMETER_LIST> element may contain at least one <PARAMETER> element.

A23 The <PARAMETER> element may have the following attributes. The NAME attribute may be the name of parameter. The TYPE attribute may be the enumeration of parameter type. The SOURCE attribute may be the name of configurable block or resource containing the bitfield associated with the parameter. The REGISTER_NAME attribute may be the name of the register containing the bitfield associated with the parameter. The BITFIELD attribute may be the name of the bitfield associated with the parameter. The ORDER attribute may be the order that the parameters appear in the list. The VALUE attribute may be the default value. The VALUE_TYPE attribute may be ENUM or INT. The default value is ENUM if attribute is missing. The MIN attribute may apply only to VALUE_TYPE = INT, which is the minimum inclusive parameter value. The MAX attribute may apply only to VALUE_TYPE = INT, which may be the maximum inclusive parameter value.

~~Please replace~~ the paragraph beginning at page 31, line 20 with the following:

A24 The SOURCE, REGISTER_NAME, and BITFIELD attributes specify the configurable block, or resource, and the bitfield, in the shape that contains the bitfield associated with the parameter. The SOURCE must be set to a NAME included in the <SHAPE> element of a <BLOCK> element, or of a <RESOURCE> element. The REGISTER_NAME and BITFIELD attributes must also be included in the <BLOCK> or <RESOURCE> element. A special keyword for the SOURCE attribute is ALL_DIGITAL. If the SOURCE for a parameter is set to ALL_DIGITAL, then the parameter applies to a similar bitfield for all configurable digital blocks defined in the User Module. This value can be used to set all clocks for the digital blocks to the same value.

~~Please replace the paragraph beginning at page 32, line 11 with the following:~~

A25
The TYPE attribute may be a UI helper that controls the appearance of parameters on the configurable blocks in the Placement Pane. When a User Module is placed on to the configurable blocks, some of the parameters may be shown on the blocks. When parameters are shown on the configurable blocks, then they may be set from the Placement Pane by clicking on the active area for the parameter on the block, as shown in the Parameter Block Selection screen. The enumerated values of the TYPE attribute determine where on the block the active area for the parameter is shown. The valid values for the TYPE attribute are described below.

VERSION OF AMENDMENTS WITH CHANGES SHOWN:

IN THE SPECIFICATION

Please replace the paragraph beginning at page 9, line 16 with the following:

This Application makes reference to co-pending commonly-owned United States Patent Application Serial No. [] 10/033,027, filed [] October 22, 2001, attorney docket number CYPR - CD00232 entitled “Programmable Microcontroller [System on a Chip] Architecture,” which is hereby incorporated herein by reference in its entirety.

Please replace the paragraph beginning at page 10, line 10 with the following:

A new class of microcontroller provides an analog and/or digital subsystem comprising many dynamically configurable blocks. An example of such a device is the [CY8C25x/26x] CY8C25xxx/26xxx family, commercially available from Cypress MicroSystems, Inc., of Bothell, Washington. These blocks do not, in general, provide the equivalent function of a traditional microprocessor peripheral. They may be thought of as fundamental building blocks. However, these blocks may be combined to create such functions. In further contrast to the fixed peripherals of a

traditional microcontroller, these blocks may be recombined in a different combination to perform a different function.

Please replace the paragraph beginning at page 15, line 10 with the following:

System 600 optionally includes a radio frequency module 660, which may implement a variety of wireless protocols, for example IEEE 802.11 or [Bluetooth] BLUETOOTH™ telecommunication and computer communications (BLUETOOTH is a trademark of Bluetooth SIG, Inc., Washington D.C.).

Please replace the paragraph beginning at page 15, line 18 with the following:

Microcontroller system 300 may contain microcontroller 350. Microcontroller 350 may execute instructions stored within memory elements of microprocessor system 300 to configure configurable elements of microcontroller system 300, including configurable analog [PSoC] blocks 310, [Digital PSoC] configurable digital blocks 320, programmable interconnect 330 and configurable I/O transceivers 340. Configuration information for these configurable elements may have been generated by embodiments of the present invention.

Please replace the paragraph beginning at page 15, line 25 with the following:

Microcontroller system 300 may include configurable analog [PSoC] (“PSOC”) blocks 310. [“PSoC”] “PSOC” is a [trade name] trademark of Cypress MicroSystems of Bothell, Washington, [for “Programmable System on (a) Chip,” and refers to an architecture of low-level building blocks] for [creating a wide variety of on-chip functions associated with] a programmable, dynamically configurable microcontroller. [Analog PSoC] Configurable analog blocks 310 may be an array of [such] low level building blocks designed to implement analog functions. Similarly, configurable

digital [PSoC] blocks 320 may be an array of low level building blocks designed to implement digital functions.

Please replace the paragraph beginning at page 17, line 22 with the following:

Figure 5 shows an exemplary User Module placement display 500, according to an embodiment of the present invention. User Module placement involves the placement of selected User Modules onto the [PSoC] configurable blocks 310 and/or 320 available on the selected device.

Please replace the paragraph beginning at page 19, line 16 with the following:

The CLOCK parameter 6040 may always map to an area in the upper left corner of the block 6070. Embodiments of the present invention may be configured so that there is only one parameter whose TYPE attribute is set to CLOCK in any one [PSoC] configurable block within any User Module. The CLOCK parameter does not have a text block similar to parameter text area 6060, but shows the triangle symbol within the block 6070.

Please replace the paragraph beginning at page 20, line 4 with the following:

The INPUT-MUX parameter 6050 is a special case parameter that may cause the line to be drawn between the top row of configurable analog [PSoC] blocks, the ANALOG-CT blocks, and the Pin Input MUX control above them. The INPUT-MUX 6050 parameter does not have a bitfield associated with it, so the BITFIELD attribute should be set to NONE. Consequently, an INPUT-MUX 6050 parameter will not appear in the Parameter Pane. The value shown next to the line is controlled by the Pin Input MUX control above. The INPUT-MUX 6050 parameter may only be used with ANALOG-CT blocks.

Please replace the paragraph beginning at page 22, line 16 with the following:

According to an embodiment of the present invention, the values of some of the attributes in user module description 700 are controlled by the names used in the Device Description XML files, while others are controlled by the elements and attributes in the User Module XML file itself. Each User Module XML file may contain the specifications for one, and only one User Module. From the root node [<PSOC_DEVICE_DB>] <DEVICE_DB> 702, the User Module XML file may contain one-and-only-one [<PSOC_USER_MODULE_LIST>] <USER_MODULE_LIST> 704 element. The [<PSOC_USER_MODULE_LIST>] <USER_MODULE_LIST> 704 element can contain a multiplicity of [<PSOC_USER_MODULE>] <USER_MODULE> elements, for example [<PSOC_USER_MODULE_LIST>] '<USER_MODULE_LIST> 706, but generally will contain only one. The [<PSOC_USER_MODULE>] <USER_MODULE> element may have the following attributes as described below.

Please replace the paragraph beginning at page 23, line 2 with the following:

The NAME attribute is the reference for the User Module used by [the Device Editor] an editor (configuration editing software). It is also the name that appears in under the icon in the User Module Catalog on the User Module Selection view.

Please replace the paragraph beginning at page 24, line 4 with the following:

The [<PSOC_USER_MODULE>] <USER_MODULE> element may contain one, and only one of each of the following elements: <SHAPE>, <PARAMETER_LIST> and <API_REGISTER_ALIAS_LIST>.

Please replace the paragraph beginning at page 24, line 8 with the following:

The <SHAPE> element, for example <SHAPE> element 707, specifies the [PSoC] configurable blocks and the resources required by the User Module, and establishes some reference names for use in other parts of the User Module description. The <SHAPE> element has only a SHAPE_TYPE attribute. The SHAPE_TYPE attribute may be set to BLOCK_LIST.

Please replace the paragraph beginning at page 24, line 18 with the following:

The <BLOCK_LIST> element describes a collection of [PSoC] configurable blocks that are connected between blocks within the <BLOCK_LIST> element. When multiple <BLOCK_LIST> elements are included within a <SHAPE> element, each <BLOCK_LIST> is placed on the [device PSoC] device's configurable blocks independently. Connections between [PSoC] configurable blocks from distinct <BLOCK_LIST> elements can exist through resources identified in the <RESOURCE_LIST> element.

Please replace the paragraph beginning at page 25, line 10 with the following:

The NAME attribute is a local name that identifies the block within the User Module description. The NAME must be unique within a User Module, but similar names can be used in different User Modules. The NAME attribute value appears in the GUI in the User Module Placement view. When a User Module is placed, the [PSoC] configurable blocks onto which the User Module is placed show the instance name of the User Module with the local name directly under it.

Please replace the paragraph beginning at page 25, line 17 with the following:

The TYPE attribute specifies the type of [PSoC] configurable block that the User Module requires. The valid values for the TYPE attributes include: DIGITAL, DIGITAL_COMM, ANALOG_CT, ANALOG_SCA, ANALOG_SCB and ANALOG_SC.

Please replace the paragraph beginning at page 25, line 22 with the following:

The difference between this list and the [PSoC] configurable block types used in the device description is the addition of the ANALOG_SC type in the User Module description. The ANALOG_SC block type indicates that the specified block can be placed on an ANALOG_SCA or an ANALOG_SCB block. Similarly, a DIGITAL block type can be placed on a DIGITAL or a DIGITAL_COMM block.

Please replace the paragraph beginning at page 26, line 1 with the following:

The INTERRUPT_SOURCE and INTERRUPT_TYPE attributes relate to an interrupt handler associated with the [PSoC] configurable block. If an interrupt handler is not associated with the [PSoC] configurable block, then the INTERRUPT_SOURCE and INTERRUPT_TYPE attributes should not be included in the <BLOCK> element. The INTERRUPT_SOURCE attribute value is a name that is used to generate the interrupt handler name. The INTERRUPT_SOURCE attribute value is appended to the User Module instance name to form the interrupt handler name. The INTERRUPT_TYPE attribute specifies whether a LJMP (long jump) or a CALL (to subroutine) is used when calling the interrupt handler from the vector table.

Please replace the paragraph beginning at page 26, line 14 with the following:

The <REGISTER_LIST> element, for example <REGISTER_LIST> element 712, may specify the [PSoC] configurable block personalization. The bitfield values set in this element are set on the [PSoC] configurable block where the User Module is located.

Please replace the paragraph beginning at page 26, line 18 with the following:

The <REGISTER_LIST> consists of one-or-more <REGISTER> elements, for example <REGISTER> element 714. The <REGISTER> element may have a NAME attribute and one <BITFIELD_LIST> element. The NAME attribute value must match one of the names of the registers in a [PSoC] configurable block of similar type contained in the <COMMON_BLOCK_LIST> element of the device description.

Please replace the paragraph beginning at page 27, line 5 with the following:

The <INPUT_LIST> element, for example <INPUT_LIST> 720 of Figure 7B, specifies the interconnection between blocks and resources within the User Module description. The <INPUT_LIST> element has no attributes and consists of one-or-more <INPUT> elements. The <INPUT> element specifies a connection between the block and another block or resource within the User Module description. The relevance of an <INPUT> element is that a connection with another [PSoC] configurable block or resource on the device implies that a bitfield within the block registers must be set to a particular value.

Please replace the paragraph beginning at page 27, line 14 with the following:

The <INPUT> element may have the attributes described below. The SOURCE attribute may be the name of the resource or [PSoC] configurable block that is the source of the input. The TYPE attribute may be an enumeration of the type of the SOURCE, either BLOCK or RESERVED_RESOURCE. The BLOCK_TYPE attribute may be an enumeration of the type of the block on which the block is placed, either ANALOG_SCA or ANALOG_SCB. The REGISTER_NAME attribute may be the name of the register within the block <REGISTER_LIST> containing the relevant bitfield. The BITFIELD attribute may be the name of the bitfield associated with the input.

Please replace the paragraph beginning at page 29, line 25 with the following:

The SOURCE attribute may be the name of the resource or [PSoC] configurable block that is the source of the input. The TYPE attribute may be an enumeration of the type of the SOURCE, either BLOCK or RESERVED_RESOURCE. The REGISTER_NAME attribute may be the name of the register within the block <REGISTER_LIST> containing the relevant bitfield. The BITFIELD attribute may be the name of the bitfield associated with the input.

Please replace the paragraph beginning at page 30, line 24 with the following:

The <PARAMETER_LIST> element may contain at least one <PARAMETER> [elements] element. The <PARAMETER> element may have the following attributes. The NAME attribute may be the name of parameter. The TYPE attribute may be the enumeration of parameter type. The SOURCE attribute may be the name of [PSoC] configurable block or resource containing the bitfield associated with the parameter. The REGISTER_NAME attribute may be the name of the register containing the bitfield associated with the parameter. The BITFIELD attribute may be the name of the bitfield associated with the parameter. The ORDER attribute may be the order that the parameters appear in the list. The VALUE attribute may be the default value. The VALUE_TYPE attribute may be ENUM or INT. The default value is ENUM if attribute is missing. The MIN attribute may apply only to VALUE_TYPE=INT, which is the minimum inclusive parameter value. The MAX attribute may apply only to VALUE_TYPE=INT, which may be the maximum inclusive parameter value.

Please replace the paragraph beginning at page 31, line 20 with the following:

The SOURCE, REGISTER_NAME, and BITFIELD attributes specify the [PSoC] configurable block, or resource, and the bitfield, in the shape that contains the bitfield [associate] associated with the parameter. The SOURCE must be set to a NAME included in the <SHAPE>

element of a <BLOCK> element, or of a <RESOURCE> element. The REGISTER_NAME and BITFIELD attributes must also be included in the <BLOCK> or <RESOURCE> element. A special keyword for the SOURCE attribute is ALL_DIGITAL. If the SOURCE for a parameter is set to ALL_DIGITAL, then the parameter applies to a similar bitfield for all configurable digital [PSoC] blocks defined in the User Module. This value can be used to set all clocks for the digital blocks to the same value.

Please replace the paragraph beginning at page 32, line 11 with the following:

The TYPE attribute may be a UI helper that controls the appearance of parameters on the [PSoC] configurable blocks in the Placement Pane. When a User Module is placed on to the [PSoC] configurable blocks, some of the parameters may be shown on the blocks. When parameters are shown on the [PSoC] configurable blocks, then they may be set from the Placement Pane by clicking on the active area for the parameter on the block, as shown in the Parameter Block Selection screen. The enumerated values of the TYPE attribute determine where on the block the active area for the parameter is shown. The valid values for the TYPE attribute are described below.

IN THE DRAWINGS

Applicant respectfully requests approval of the drawing changes proposed in the enclosed Request for Approval of Drawing Changes.

SUPPORT FOR AMENDMENTS

Support for the amendments herein can be found in the specification and Figures as originally filed (e.g., page 2, lines 20-23; page 9, lines 16-20; page 10, lines 10-24; page 15, line 25 - page 16, line 4; and Figures 4 and 5), in the related application Ser. No. 10/033,027, and in Exhibits A-C attached hereto (containing information relating to trademarks and generic product numbers